

# Two Dimensional Angle of Arrival Estimation

Santhosh Kumar, Pradip Sircar

Department of Electrical Engineering

Indian Institute of Technology Kanpur

Kanpur 208016, Uttar Pradesh, India

## 1 Introduction

We present a new method for the estimation of two dimensional (2D) angles of arrival (AOAs), namely, azimuth and incidence angles of multiple narrowband signals of same frequency in the far field of antenna array. In this report, we propose an algorithm which uses an L-shape array configuration [1]. We extend the polynomial based approach presented in [2] to two dimensional AOA estimation.

We Consider an L-shaped antenna array as shown in Figure 1 in the XZ plane [1].

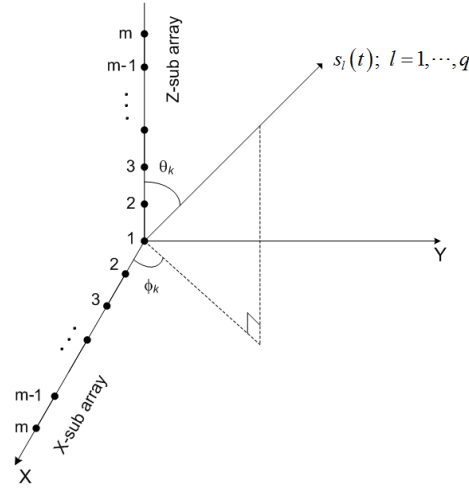


Figure 1: The L-shaped array configuration

The gain of an element at a position  $(x, y, z)$  for a signal with reference to

origin from the direction  $(\theta, \phi)$  is given by

$$a(\theta, \phi) = e^{j(2\pi/\lambda)(x \sin \theta \cos \phi + y \sin \theta \sin \phi + z \cos \theta)} \quad (1)$$

Suppose that there are  $q$  narrow band sources, with same wavelength  $\lambda$  impinging on each sensor. An  $l$ th source has an incidence angle  $\theta_l$  and an azimuth angle  $\phi_l$ ,  $l = 1, 2, \dots, q$ .

The signal received at the  $i$ th element of the Z-subarray is given by [1]

$$z_i(t) = \sum_{l=1}^q a_{z,i}(\theta_l, \phi_l) s_l(t) + n_{z,i}(t) \quad (2)$$

where

$$\begin{aligned} a_{z,i}(\theta_l, \phi_l) &= e^{j\beta d \cos \theta_l} \\ &= e^{j2\pi(i-1)(d/\lambda) \cos \theta_l} \\ &= e^{j(i-1)\psi_l} \end{aligned} \quad (3)$$

$$\psi_l = 2\pi(d/\lambda) \cos \theta_l$$

$s_l(t)$  is the complex envelope of the  $l$ th signal arriving from the direction  $(\theta_l, \phi_l)$  and  $n_{z,i}(t)$  is the zero-mean additive white Gaussian noise (AWGN) whose properties are mentioned in Section 2.2.

The signal received at the  $i$ th element of the X-subarray is given by

$$x_i(t) = \sum_{l=1}^q a_{x,i}(\theta_l, \phi_l) s_l(t) + n_{x,i}(t) \quad (4)$$

where

$$\begin{aligned} a_{x,i}(\theta_l, \phi_l) &= e^{j2\pi(i-1)(d/\lambda) \sin \theta_l \cos \phi_l} \\ &= e^{j(i-1)\xi_l}, \end{aligned} \quad (5)$$

$$\xi_l = 2\pi(d/\lambda) \sin \theta_l \cos \phi_l$$

and  $n_{x,i}(t)$  is the zero-mean AWGN.

If we collect outputs of all the sensors on the Z-subarray into an  $m \times 1$  vector, we have

$$\mathbf{z}(t) = \sum_{l=1}^q \mathbf{a}_z(\theta_l, \phi_l) s_l(t) + \mathbf{n}_z(t) \quad (6)$$

where

$$\mathbf{z}(t) = [z_1(t) \quad z_2(t) \quad \cdots \quad z_m(t)]^T$$

$\mathbf{a}_z(\theta_l, \phi_l)$  is the  $m \times 1$  steering vector of the Z-subarray towards the direction  $(\theta_l, \phi_l)$  given by

$$\mathbf{a}_z(\theta_l, \phi_l) = [a_{z,1}(\theta_l, \phi_l) \quad a_{z,2}(\theta_l, \phi_l) \quad \cdots \quad a_{z,m}(\theta_l, \phi_l)]^T$$

and

$$\mathbf{n}_z(t) = [n_{z,1}(t) \quad n_{z,2}(t) \quad \cdots \quad n_{z,m}(t)]^T$$

We can rewrite (6) as

$$\mathbf{z}(t) = \begin{pmatrix} a_{z,1}(\theta_1, \phi_1) & a_{z,1}(\theta_2, \phi_2) & \cdots & a_{z,1}(\theta_q, \phi_q) \\ a_{z,2}(\theta_1, \phi_1) & a_{z,2}(\theta_2, \phi_2) & \cdots & a_{z,2}(\theta_q, \phi_q) \\ \vdots & \vdots & \vdots & \vdots \\ a_{z,m}(\theta_1, \phi_1) & a_{z,m}(\theta_2, \phi_2) & \cdots & a_{z,m}(\theta_q, \phi_q) \end{pmatrix} \begin{pmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_q(t) \end{pmatrix} + \begin{pmatrix} n_{z,1}(t) \\ n_{z,2}(t) \\ \vdots \\ n_{z,m}(t) \end{pmatrix} \quad (7)$$

Similarly for the X-subarray, we have

$$\mathbf{x}(t) = \sum_{l=1}^q \mathbf{a}_x(\theta_l, \phi_l) s_l(t) + \mathbf{n}_x(t) \quad (8)$$

where

$$\mathbf{x}(t) = [x_1(t) \quad x_2(t) \quad \cdots \quad x_m(t)]^T$$

$$\mathbf{a}_x(\theta_l, \phi_l) = [a_{x,1}(\theta_l, \phi_l) \quad a_{x,2}(\theta_l, \phi_l) \quad \cdots \quad a_{x,m}(\theta_l, \phi_l)]^T$$

and

$$\mathbf{n}_x(t) = [n_{x,1}(t) \quad n_{x,2}(t) \quad \cdots \quad n_{x,m}(t)]^T$$

Note that (8) can be rewritten as

$$\mathbf{x}(t) = \begin{pmatrix} a_{x,1}(\theta_1, \phi_1) & a_{x,1}(\theta_2, \phi_2) & \cdots & a_{x,1}(\theta_q, \phi_q) \\ a_{x,2}(\theta_1, \phi_1) & a_{x,2}(\theta_2, \phi_2) & \cdots & a_{x,2}(\theta_q, \phi_q) \\ \vdots & \vdots & \vdots & \vdots \\ a_{x,m}(\theta_1, \phi_1) & a_{x,m}(\theta_2, \phi_2) & \cdots & a_{x,m}(\theta_q, \phi_q) \end{pmatrix} \begin{pmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_q(t) \end{pmatrix} + \begin{pmatrix} n_{x,1}(t) \\ n_{x,2}(t) \\ \vdots \\ n_{x,m}(t) \end{pmatrix} \quad (9)$$

## 2 Proposed Method

Consider the Z-subarray:

On substituting the values of  $a_{z,i}(\theta_l, \phi_l)$  in (7) for  $i = 1, 2, \dots, m$  and  $l = 1, 2, \dots, q$ , we have

$$\mathbf{z}(t) = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ e^{j\psi_1} & e^{j\psi_2} & \cdots & e^{j\psi_q} \\ \vdots & \vdots & \vdots & \vdots \\ e^{j(m-1)\psi_1} & e^{j(m-1)\psi_2} & \cdots & e^{j(m-1)\psi_q} \end{pmatrix} \begin{pmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_q(t) \end{pmatrix} + \begin{pmatrix} n_{z,1}(t) \\ n_{z,2}(t) \\ \vdots \\ n_{z,m}(t) \end{pmatrix} \quad (10)$$

where  $\{\psi_l\}$  are defined in (3).

By collecting  $M$  equispaced snapshots from each sensor, a data matrix  $Z$  is formed as follows

$$Z = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ e^{j\psi_1} & e^{j\psi_2} & \cdots & e^{j\psi_q} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j(m-1)\psi_1} & e^{j(m-1)\psi_2} & \cdots & e^{j(m-1)\psi_q} \end{pmatrix} \begin{pmatrix} s_1(t_1) & s_1(t_2) & \cdots & s_1(t_M) \\ s_2(t_1) & s_2(t_2) & \cdots & s_2(t_M) \\ \vdots & \vdots & \ddots & \vdots \\ s_q(t_1) & s_q(t_2) & \cdots & s_q(t_M) \end{pmatrix} + \begin{pmatrix} n_{z,1}(t_1) & n_{z,1}(t_2) & \cdots & n_{z,1}(t_M) \\ n_{z,2}(t_1) & n_{z,2}(t_2) & \cdots & n_{z,2}(t_M) \\ \vdots & \vdots & \ddots & \vdots \\ n_{z,m}(t_1) & n_{z,m}(t_2) & \cdots & n_{z,m}(t_M) \end{pmatrix} \quad (11)$$

or

$$Z = A_z S + N_z \quad (12)$$

where

$$Z = \begin{pmatrix} z_1(t_1) & z_1(t_2) & \cdots & z_1(t_M) \\ z_2(t_1) & z_2(t_2) & \cdots & z_2(t_M) \\ \vdots & \vdots & \ddots & \vdots \\ z_m(t_1) & z_m(t_2) & \cdots & z_m(t_M) \end{pmatrix}$$

The matrices  $A_z$  and  $S$  are unknown, and are not rank-deficient by assumption.

Now, let  $y_1 = e^{j\psi_1}$ ,  $y_2 = e^{j\psi_2}$ , ...,  $y_q = e^{j\psi_q}$  and assume that  $y_1, y_2, \dots, y_q$  are the roots of a polynomial equation

$$\mathcal{P}(y) = 1 + c_1 y + c_2 y^2 + \cdots + c_{m-1} y^{m-1} = 0. \quad (13)$$

Now our aim is to calculate the coefficients which satisfy this assumption and then to solve for the roots.

## 2.1 Method without Noise

From (13) and (11), it can easily be shown that the following equations hold for no noise case.

$$\begin{aligned} z_1(t_1) + c_1 z_2(t_1) + c_2 z_3(t_1) + \cdots + c_{m-1} z_m(t_1) &= 0 \\ z_1(t_2) + c_1 z_2(t_2) + c_2 z_3(t_2) + \cdots + c_{m-1} z_m(t_2) &= 0 \\ \vdots & \\ z_1(t_M) + c_1 z_2(t_M) + c_2 z_3(t_M) + \cdots + c_{m-1} z_m(t_M) &= 0 \end{aligned}$$

After simple manipulations,

$$\begin{aligned} c_1 z_2(t_1) + c_2 z_3(t_1) + \cdots + c_{m-1} z_m(t_1) &= -z_1(t_1) \\ c_1 z_2(t_2) + c_2 z_3(t_2) + \cdots + c_{m-1} z_m(t_2) &= -z_1(t_2) \\ \vdots & \\ c_1 z_2(t_M) + c_2 z_3(t_M) + \cdots + c_{m-1} z_m(t_M) &= -z_1(t_M) \end{aligned}$$

The above set of equations can be written in matrix form as

$$\begin{pmatrix} z_2(t_1) & z_3(t_1) & \cdots & z_m(t_1) \\ z_2(t_2) & z_3(t_2) & \cdots & z_m(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ z_2(t_M) & z_3(t_M) & \cdots & z_m(t_M) \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{m-1} \end{pmatrix} = \begin{pmatrix} -z_1(t_1) \\ -z_1(t_2) \\ \vdots \\ -z_1(t_M) \end{pmatrix} \quad (14)$$

or

$$PC = P_1 \quad (15)$$

This is an overdetermined system of equations. The best solution in the least square sense for the coefficient vector  $C$  is given by

$$C = P^\# P_1 \quad (16)$$

where  $P^\#$  is the Moore-Penrose pseudoinverse of  $P$ ,

$$P^\# = (P^H P)^{-1} P^H$$

Substitute  $c_1, c_2, \dots, c_{m-1}$  obtained in (16) into (13) and solve for the roots of  $\mathcal{P}(y)$ . There are  $(m-1)$  roots, and among those only  $q$  solutions are of our interest. From the  $(m-1)$  roots choose  $q$  roots say  $\{\gamma_l\}$ , whose magnitudes are nearer to unity:

$$\gamma_l = e^{j2\pi(d/\lambda) \cos \theta_l} \quad l = 1, 2, \dots, q \quad (17)$$

Then, the estimate of  $\psi_l$  is

$$\hat{\psi}_l = \text{angle}\{\gamma_l\} \quad (18)$$

Follow the similar approach for the X-subarray to calculate  $\{\beta_l\}$  defined by

$$\beta_l = e^{j2\pi(d/\lambda) \sin \theta_l \cos \phi_l} \quad l = 1, 2, \dots, q \quad (19)$$

and hence

$$\hat{\xi}_l = \text{angle}\{\beta_l\} \quad (20)$$

Now the estimates of  $\theta_l$  and  $\phi_l$  are given by

$$\hat{\theta}_l = \sin^{-1} \left( \frac{\hat{\psi}_l}{2\pi(d/\lambda)} \right) \quad l = 1, 2, \dots, q. \quad (21)$$

$$\hat{\phi}_l = \cos^{-1} \left( \frac{\hat{\xi}_l}{(2\pi(d/\lambda) \sin \hat{\theta}_l)} \right) \quad l = 1, 2, \dots, q. \quad (22)$$

## 2.2 Method with Noise

Note that in (6) and (8), the vectors  $\mathbf{n}_z(t)$  and  $\mathbf{n}_x(t)$  are random noise components on Z and X subarrays respectively. The noise is assumed to be additive, temporally white Gaussian with zero mean. This assumption is a valid assumption in real-world applications.

Mathematically, we can express the above stated properties of the noise as follows,

$$E[\mathbf{n}_z(t)] = E[\mathbf{n}_x(t)] = \mathbf{0} \quad (23)$$

$$E[\mathbf{n}_z(t)\mathbf{n}_z^H(t)] = E[\mathbf{n}_x(t)\mathbf{n}_x^H(t)] = \sigma^2\mathbf{I} \quad (24)$$

$$E[\mathbf{n}_z(t)\mathbf{n}_z^T(t)] = E[\mathbf{n}_x(t)\mathbf{n}_x^T(t)] = \mathbf{0} \quad (25)$$

and

$$E[\mathbf{n}_z(t_i)\mathbf{n}_z^H(t_j)] = E[\mathbf{n}_z(t_i)\mathbf{n}_z^T(t_j)] = \mathbf{0} \quad \text{for } i \neq j \quad (26)$$

$$E[\mathbf{n}_x(t_i)\mathbf{n}_x^H(t_j)] = E[\mathbf{n}_x(t_i)\mathbf{n}_x^T(t_j)] = \mathbf{0} \quad \text{for } i \neq j \quad (27)$$

where  $E$  denotes the expectation operation,  $^H$  denotes the Hermitian operation,  $^T$  denotes the transposition, and  $t_i$  and  $t_j$  are the  $i$ th and  $j$ th time instants respectively.

The noise has a negative impact on the accuracy of estimates of the polynomial coefficients. In order to reduce the effect of the noise, we follow the same procedure except for the calculation of inverse of the matrix  $P$  in (15) where we use the singular value decomposition (SVD) as discussed below.

Let the SVD of  $P$  be

$$P = U\Sigma V^H \quad (28)$$

where  $U$  and  $V$  are the unitary matrices containing the singular vectors of  $P$  and  $\Sigma$  is a diagonal matrix of same dimensions as  $P$ , and it is given by

$$\Sigma = U^H P V \quad (29)$$

The diagonal elements in  $\Sigma$  are non-negative and are sorted in decreasing order, i.e.,

$$\Sigma = \text{diag}\{\tau_1, \tau_2, \dots, \tau_q, \dots, \tau_{m-1}\} \quad (30)$$

where

$$\tau_1 \geq \tau_2 \geq \dots \geq \tau_q \geq \dots \geq \tau_{m-1}$$

are the singular values of  $P$ .

To reduce the effect of noise, replace the last  $m - q - 1$  singular values with zero in calculating the inverse of  $P$ , i.e.,

$$\Sigma_1 = \text{diag}\{\tau_1, \tau_2, \dots, \tau_q, 0, \dots\} \quad (31)$$

and

$$\Sigma_1^{-1} = \text{diag}\{1/\tau_1, 1/\tau_2, \dots, 1/\tau_q, 0, \dots\} \quad (32)$$

The dimensions of  $\Sigma_1^{-1}$  are same as that of  $P^H$ .  
The least squares approximation of  $P^{-1}$  is given by [3]

$$P^{-1} = V\Sigma_1^{-1}U^H \quad (33)$$

hence, the estimate of coefficient vector in (15) is

$$C = V\Sigma_1^{-1}U^H P_1 \quad (34)$$

Follow the same approach for X-subarray to obtain the coefficient vector and the roots. The computation of AOAs from the roots is same as that for the noiseless case.

## References

- [1] N. Tayem and H. M. Kwon, "L-shape 2-dimensional arrival angle estimation with propagator method," *IEEE Trans. Antennas and Propagation*, vol. 53, no. 5, pp. 1622–1630, May 2005.
- [2] P. Singh and P. Sircar, "Time delays and angles of arrival estimation using known signals", *Signal, Image and Video Processing*, vol. 6, no. 2, pp. 171–178, Feb. 2011.
- [3] P. Dewilde and E. F. Deprettere, "Singular value decomposition: An introduction," In E. F. Deprettere (ed.), *SVD and Signal Processing: Algorithms, Applications, and Architectures*, pp. 3–41. Elsevier Science, North Holland, 1988.